STATE OF ALABAMA

Information Technology Guideline

Guideline 660-02G5: Security Engineering Principles

1. INTRODUCTION:

As security awareness becomes a way of life within an organization, people at all levels, and roles in the system life-cycle, should have a basic understanding of the security principles governing the systems they are using, maintaining, or designing and developing.

To aid in designing secure information systems, the National Institute of Standards and Technology (NIST) compiled a set of engineering principles for system security. These principles provide a foundation upon which a more consistent and structured approach to the design, development, and implementation of IT security capabilities can be constructed.

While the primary focus of these principles is the implementation of technical controls, these principles highlight the fact that, to be effective, a system security design should also consider non-technical issues, such as policy, operational procedures, and user awareness and training.

Ideally, the principles presented here would be used from the onset of a program then employed throughout the system's lifecycle. However, these principles are also helpful in affirming or confirming the security posture of already deployed information systems. The principles are short and concise and can be used by all organizations to develop their system life-cycle policies.

2. OBJECTIVE:

Design, develop, and operate information systems using security engineering principles.

3. SCOPE:

This guideline presents generic security engineering principles that can be applied to all systems (but not at all times because of the constantly changing information system security environment; each principle should be carefully considered throughout the life-cycle of every system). The principles presented herein can be used by:

- Users when developing and evaluating functional requirements, or when operating information systems within their organizations.
- System Engineers and Architects when designing, implementing, or modifying an information system.
- IT Specialists during all phases of the system life-cycle.
- Program Managers and Information Security Officers to ensure adequate security measures have been considered for all phases of the system life-cycle.

4. GUIDELINES:

The application of security engineering principles is primarily targeted at new information systems under development or systems undergoing major upgrades and should be integrated into the system development life cycle. For legacy information systems, organizations should apply security engineering principles to system upgrades and modifications, to the extent feasible, given the current state of the hardware, software, and firmware components within the system.

As recommended by NIST in Special Publication 800-27: Engineering Principles for Information Technology Security, the following system-level security principles should be utilized in the design, development, and operation of State of Alabama information systems.

4.1 SECURITY FOUNDATION

Principle 1: Establish a sound security policy as the "foundation" for design.

> A security policy is an important document to develop while designing an information system. The security policy begins with the organization's basic commitment to information security formulated as a general policy statement. The policy is then applied to all aspects of the system design or security solution. The policy identifies security goals (e.g., confidentiality, integrity, availability, accountability, and assurance) the system should support and these goals guide the procedures, standards and controls used in the IT security architecture design. The policy also should require definition of critical assets, the perceived threat, and security-related roles and responsibilities.

Principle 2: Treat security as an integral part of the overall system design.

> Security must be considered in information system design and should be integrated fully into the system life-cycle. Experience has shown it to be both difficult and costly to introduce security measures properly and successfully after a system has been developed, so security should be implemented in the design stage of all new information systems, and where possible, in the modification and continuing operation of all legacy systems. This includes establishing security policies, understanding the resulting security requirements, participating in the evaluation of security products, and in the engineering, design, implementation, and disposal of the system.

Principle 3: Clearly delineate the physical and logical security boundaries governed by associated security policies.

> Information technology exists in physical and logical locations, and boundaries exist between these locations. An understanding of what is to be protected from external factors can help ensure adequate protective measures are applied where they will be most effective.
>
> Sometimes a boundary is defined by people, information, and information technology associated with one physical location. But this ignores the reality that, within a single location, many different security policies may be in place, some covering publicly accessible information and some covering sensitive or confidential information. Other times a boundary is defined by a security policy that governs a specific set of information and information technology that can cross physical boundaries. Further complicating the

matter is that, many times, a single machine or server may house both public-access and sensitive information. As a result, multiple security policies may apply to a single machine or within a single system. Therefore, when developing an information system, security boundaries must be considered and communicated in relevant system documentation and security policies.

Principle 4: Ensure that developers are trained in how to develop secure software.

Ensure that developers are adequately trained in the design, development, configuration control, integration, and testing of secure software before developing the system.

4.2 RISK BASED

Principle 5: Reduce risk to an acceptable level.

Risk is defined as the combination of (1) the likelihood that a particular threat source will exercise (intentionally exploit or unintentionally trigger) a particular information system vulnerability and (2) the resulting adverse impact on organizational operations, assets, or individuals should this occur.

Recognize that the elimination of all risk is not cost-effective. A cost-benefit analysis should be conducted for each proposed control. In some cases, the benefits of a more secure system may not justify the direct and indirect costs. Benefits include more than just prevention of monetary loss; for example, controls may be essential for maintaining public trust and confidence. Direct costs include the cost of purchasing and installing a given technology; indirect costs include decreased system performance and additional training. The goal is to enhance mission/business capabilities by mitigating mission/business risk to an acceptable level. (Related Principle: 6)

Principle 6: Assume that external systems are insecure.

The term information domain arises from the practice of partitioning information resources according to access control, need, and levels of protection required. Organizations implement specific measures to enforce this partitioning and to provide for the deliberate flow of authorized information between information domains. The boundary of an information domain represents the security perimeter for that domain.

An external domain is one that is not under your control. In general, external systems should be considered insecure. Until an external domain has been deemed “trusted,” system engineers, architects, and IT specialists should presume the security measures of an external system are different than those of a trusted internal system and design the system security features accordingly.

Principle 7: Identify potential trade-offs between reducing risk and increased costs and decrease in other aspects of operational effectiveness.

To meet stated security requirements, a systems designer, architect, or security practitioner will need to identify and address all competing operational needs. It may be necessary to modify or adjust security goals due to other operational requirements. In modifying or adjusting security goals, an acceptance of greater risk and cost may be inevitable. By identifying and addressing these trade-offs as early as possible, decision

makers will have greater latitude and be able to achieve more effective systems. (Related Principle: 4)

Principle 8: Implement tailored system security measures to meet organizational security goals.

In general, IT security measures are tailored according to an organization's unique needs. While numerous factors, such as the overriding mission requirements, and guidance, are to be considered, the fundamental issue is the protection of the mission or business from IT security-related, negative impacts. Because IT security needs are not uniform, system designers and security practitioners should consider the level of trust when connecting to other external networks and internal sub-domains. Recognizing the uniqueness of each system allows a layered security strategy to be used – implementing lower assurance solutions with lower costs to protect less critical systems and higher assurance solutions only at the most critical areas.

Principle 9: Protect information while being processed, in transit, and in storage.

The risk of unauthorized modification or destruction of data, disclosure of information, and denial of access to data while in transit should be considered along with the risks associated with data that is in storage or being processed. Therefore, system engineers, architects, and IT specialists should implement security measures to preserve, as needed, the integrity, confidentiality, and availability of data, including application software, while the information is being processed, in transit, and in storage.

Principle 10: Consider custom products to achieve adequate security.

Designers should recognize that in some instances it may not be possible to meet security goals with systems constructed entirely from commercial off-the-shelf (COTS) products. In such instances, it may be necessary to augment COTS with non-COTS mechanisms.

Principle 11: Protect against all likely classes of "attacks."

In designing the security controls, multiple classes of "attacks" need to be considered. Those classes that result in unacceptable risk need to be mitigated. Examples of "attack" classes are: passive monitoring, active network attacks, exploitation by insiders, attacks requiring physical access or proximity, and the insertion of backdoors and malicious code during software development and/or distribution.

4.3 EASE OF USE

Principle 12: Where possible, base security on open standards for portability and interoperability.

Most organizations depend significantly on distributed information systems to perform their mission or business. These systems distribute information both across their own organization and to other organizations. For security capabilities to be effective in such environments, security program designers should make every effort to incorporate interoperability and portability into all security measures, including hardware and software, and implementation practices.

Principle 13: Use common language in developing security requirements.

The use of a common language when developing security requirements permits organizations to evaluate and compare security products and features evaluated in a common test environment. When a “common” evaluation process is based upon common requirements or criteria, a level of confidence can be established that ensures product security functions conform to an organization’s security requirements.

The Common Criteria (CC; available at http://www.commoncriteriaportal.org/) provides a source of common expressions for common needs and supports a common assessment methodology. Use of CC "protection profiles" and "security targets" greatly aids the development of products (and to some extent systems) that have IT security functions. The rigor and repeatability of the CC methodology provides for thorough definition of user security needs. Security targets provide system integrators with key information needed in the procurement of components and implementation of secure IT systems.

Principle 14: Design security to allow for regular adoption of new technology, including a secure and logical technology upgrade process.

As mission and business processes and the threat environment change, security requirements and technical protection methods must be updated. IT-related risks to the mission/business vary over time and undergo periodic assessment. Periodic assessment should be performed to enable system designers and managers to make informed risk management decisions on whether to accept or mitigate identified risks with changes or updates to the security capability. The lack of timely identification through consistent security solution re-evaluation and correction of evolving, applicable IT vulnerabilities results in false trust and increased risk.

Each security mechanism should be able to support migration to new technology or upgrade of new features without requiring an entire system redesign. The security design should be modular so that individual parts of the security design can be upgraded without the requirement to modify the entire system.

Principle 15: Strive for operational ease of use.

The more difficult it is to maintain and operate a security control the less effective that control is likely to be. Therefore, security controls should be designed to be consistent with the concept of operations and with ease-of-use as an important consideration. The experience and expertise of administrators and users should be appropriate and proportional to the operation of the security control. An organization must invest the resources necessary to ensure system administrators and users are properly trained. Moreover, administrator and user training costs along with the life-cycle operational costs should be considered when determining the cost-effectiveness of the security control.

4.4 INCREASE RESILIENCE

Principle 16: Implement layered security (Ensure no single point of vulnerability).

Security designs should consider a layered approach to address or protect against a specific threat or to reduce vulnerability. For example, the use of a packet-filtering router in conjunction with an application gateway and an intrusion detection system combine to

increase the work-factor an attacker must expend to successfully attack the system. Add good password controls and adequate user training to improve the system's security posture even more.

By using multiple, overlapping protection approaches, the failure or circumvention of any individual protection approach will not leave the system unprotected. Through user training and awareness, well-crafted policies and procedures, and redundancy of protection mechanisms, layered protections enable effective protection of information technology for the purpose of achieving mission objectives.

The need for layered protections is especially important when COTS products are used. Practical experience has shown that the current state-of-the-art for security quality in COTS products does not provide a high degree of protection against sophisticated attacks. It is possible to help mitigate this situation by placing several controls in series, requiring additional work by attackers to accomplish their goals.

Principle 17: Design and operate an IT system to limit damage and to be resilient in response.

Information systems should be resistant to attack, should limit damage, and should recover rapidly when attacks do occur. The principle suggested here recognizes the need for adequate protection technologies at all levels to ensure that any potential cyber attack will be countered effectively. There are vulnerabilities that cannot be fixed, those that have not yet been fixed, those that are not known, and those that could be fixed but are not (e.g., risky services allowed through firewalls) to allow increased operational capabilities. In addition to achieving a secure initial state, secure systems should have a well-defined status after failure, either to a secure failure state or via a recovery procedure to a known secure state. Organizations should establish detect and respond capabilities, manage single points of failure in their systems, and implement a reporting and response strategy. (Related Principle: 14)

Principle 18: Provide assurance that the system is, and continues to be, resilient in the face of expected threats.

Assurance is the grounds for confidence that a system meets its security expectations. These expectations can typically be summarized as providing sufficient resistance to both direct penetration and attempts to circumvent security controls. Good understanding of the threat environment, evaluation of requirement sets, hardware and software engineering disciplines, and product and system evaluations are primary measures used to achieve assurance. Additionally, the documentation of the specific and evolving threats is important in making timely adjustments in applied security and strategically supporting incremental security enhancements.

Principle 19: Limit or contain vulnerabilities.

Design systems to limit or contain vulnerabilities. If a vulnerability does exist, damage can be limited or contained, allowing other information system elements to function properly. Limiting and containing insecurities also helps to focus response and reconstitution efforts to information system areas most in need. (Related Principle: 10)

Principle 20: Isolate public access systems from mission critical resources (e.g., data, processes, etc.).

While the trend toward shared infrastructure has considerable merit in many cases, it is not universally applicable. In cases where the sensitivity or criticality of the information is high, organizations may want to limit the number of systems on which that data is stored and isolate them, either physically or logically. Physical isolation may include ensuring that no physical connection exists between an organization's public access information resources and an organization's critical information. When implementing logical isolation solutions, layers of security services and mechanisms should be established between public systems and secure systems responsible for protecting mission critical resources. Security layers may include using network architecture designs such as demilitarized zones and screened subnets. Finally, system designers and administrators should enforce organizational security policies and procedures regarding use of public access systems.

Principle 21: Use boundary mechanisms to separate computing systems and network infrastructures.

To control the flow of information and access across network boundaries in computing and communications infrastructures, and to enforce the proper separation of user groups, a suite of access control devices and accompanying access control policies should be used. Determine the following for communications across network boundaries:

- What external interfaces are required
- Whether information is pushed or pulled
- What ports, protocols, and network services are required
- What requirements exist for system information exchanges; for example, trust relationships, database replication services, and domain name resolution processes

Principle 22: Design and implement audit mechanisms to detect unauthorized use and to support incident investigations.

Organizations should monitor, record, and periodically review audit logs to identify unauthorized use and to ensure system resources are functioning properly. In some cases, organizations may be required to disclose information obtained through auditing mechanisms to appropriate third parties, including law enforcement authorities or Freedom of Information Act (FOIA) applicants. Many organizations have implemented consent to monitor policies which state that evidence of unauthorized use (e.g., audit trails) may be used to support administrative or criminal investigations.

Principle 23: Develop and exercise contingency or disaster recovery procedures to ensure appropriate availability.

Continuity of operations plans or disaster recovery procedures address continuance of an organization's operation in the event of a disaster or prolonged service interruption that affects the organization's mission. Such plans should address an emergency response phase, a recovery phase, and a return to normal operation phase. Personnel responsibilities during an incident and available resources should be identified. In reality,

contingency and disaster recovery plans do not address every possible scenario or assumption. Rather, focus on the events most likely to occur and identify an acceptable method of recovery. Periodically, the plans and procedures should be exercised to ensure that they are effective and well understood.

4.5 REDUCE VULNERABILITIES

Principle 24: Strive for simplicity.

The more complex the mechanism, the more likely it may possess exploitable flaws. Simple mechanisms tend to have fewer exploitable flaws and require less maintenance. Further, because configuration management issues are simplified, updating or replacing a simple mechanism becomes a less intensive process.

Principle 25: Minimize the system elements to be trusted.

Security measures include people, operations, and technology. Where technology is used, hardware, firmware, and software should be designed and implemented so that a minimum number of system elements need to be trusted in order to maintain protection. Further, to ensure cost-effective and timely certification of system security features, it is important to minimize the amount of software and hardware expected to provide the most secure functions for the system.

Principle 26: Implement least privilege.

The concept of limiting access, or "least privilege," is simply to provide no more authorizations than necessary to perform required functions. This is perhaps most often applied in the administration of the system. Its goal is to reduce risk by limiting the number of people with access to critical system security controls (i.e., controlling who is allowed to enable or disable system security features or change the privileges of users or programs). Best practice suggests it is better to have several administrators with limited access to security resources rather than one person with "super user" permissions. .

Consideration should be given to implementing role-based access controls for various aspects of system use, not only administration. The system security policy can identify and define the various roles of users or processes. Each role is assigned those permissions needed to perform its functions. Each permission specifies a permitted access to a particular resource (such as "read" and "write" access to a specified file or directory, "connect" access to a given host and port, etc.). Unless permission is granted explicitly, the user or process should not be able to access the protected resource. Additionally, identify the roles/responsibilities that, for security purposes, should remain separate (this is commonly termed “separation of duties”).

Principle 27: Do not implement unnecessary security mechanisms.

Every security mechanism should support a security service or set of services, and every security service should support one or more security goals. Extra measures should not be implemented if they do not support a recognized service or security goal. Such mechanisms could add unneeded complexity to the system and are potential sources of additional vulnerabilities.

An example is file encryption supporting the access control service that in turn supports the goals of confidentiality and integrity by preventing unauthorized file access. If file encryption is a necessary part of accomplishing the goals, then the mechanism is appropriate. However, if these security goals are adequately supported without inclusion of file encryption, then that mechanism would be an unneeded system complexity.

Principle 28: Ensure proper security in the shutdown or disposal of a system.

Although a system may be powered down, critical information still resides on the system and could be retrieved by an unauthorized user or organization. Access to critical information systems must be controlled at all times.

At the end of a system's life-cycle, system designers should develop procedures to dispose of an information system's assets in a proper and secure fashion. Procedures must be implemented to ensure system hard drives, volatile memory, and other media are purged to an acceptable level and do not retain residual information.

Principle 29: Identify and prevent common errors and vulnerabilities.

Many errors reoccur with disturbing regularity - errors such as buffer overflows, race conditions, format string errors, failing to check input for validity, and programs being given excessive privileges. Learning from the past will improve future results.

4.6 DESIGN WITH THE NETWORK IN MIND

Principle 30: Implement security through a combination of measures distributed physically and logically.

Often, a single security service is achieved by cooperating elements existing on separate machines. For example, system authentication is typically accomplished using elements ranging from the user-interface on a workstation through the networking elements to an application on an authentication server. It is important to associate all elements with the security service they provide. These components are likely to be shared across systems to achieve security as infrastructure resources come under more senior budget and operational control.

Principle 31: Formulate security measures to address multiple overlapping information domains.

An information domain is a set of active entities (person, process, or devices) and their data objects. A single information domain may be subject to multiple security policies. A single security policy may span multiple information domains. An efficient and cost effective security capability should be able to enforce multiple security policies to protect multiple information domains without the need to separate physically the information and respective information systems processing the data. This principle argues for moving away from the traditional practice of creating separate LANs and infrastructures for various sensitivity levels (e.g., security classification or business function such as proposal development) and moving toward solutions that enable the use of common, shared, infrastructures with appropriate protections at the operating system, application, and workstation level.

Moreover, to accomplish missions and protect critical functions, organizations have many types of information to safeguard. With this principle in mind, system engineers, architects, and IT specialists should develop a security capability that allows organizations with multiple levels of information sensitivity to achieve the basic security goals in an efficient manner.

Principle 32: Authenticate users and processes to ensure appropriate access control decisions both within and across domains.

Authentication is the process where a system establishes the validity of a transmission, message, or a means of verifying the eligibility of an individual, process, or machine to carry out a desired action, thereby ensuring that security is not compromised by an untrusted source. It is essential that adequate authentication be achieved in order to implement security policies and achieve security goals. Additionally, level of trust is always an issue when dealing with cross-domain interactions. The solution is to establish an authentication policy and apply it to cross-domain interactions as required. Note: A user may have rights to use more than one name in multiple domains. Further, rights may differ among the domains, potentially leading to security policy violations.

Principle 33: Use unique identities to ensure accountability.

An identity may represent an actual user or a process with its own identity, e.g., a program making a remote access. Unique identities are a required element in order to be able to:

- Maintain accountability and traceability of a user or process
- Assign specific rights to an individual user or process
- Provide for non-repudiation
- Enforce access control decisions
- Establish the identity of a peer in a secure communications path
- Prevent unauthorized users from masquerading as an authorized user.

5. DEFINITIONS:

ACCESS CONTROL: Enable authorized use of a resource while preventing unauthorized use or use in an unauthorized manner.

ACCOUNTABILITY: The security goal that generates the requirement for actions of an entity to be traced uniquely to that entity. This supports non-repudiation, deterrence, fault isolation, intrusion detection and prevention, and after-action recovery and legal action.

ASSURANCE: Grounds for confidence that the other four security goals (integrity, availability, confidentiality, and accountability) have been adequately met by a specific implementation. “Adequately met” includes (1) functionality that performs correctly, (2) sufficient protection against unintentional errors (by users or software), and (3) sufficient resistance to intentional penetration or by-pass.

AUTHENTICATION: Verifying the identity of a user, process, or device, often as a prerequisite to allowing access to resources in a system.

AUTHORIZATION: The granting or denying of access rights to a user, program, or process.

AVAILABILITY: The security goal that generates the requirement for protection against intentional or accidental attempts to (1) perform unauthorized deletion of data or (2) otherwise cause a denial of service or data.

CONFIDENTIALITY: The security goal that generates the requirement for protection from intentional or accidental attempts to perform unauthorized data reads. Confidentiality covers data in storage, during processing, and while in transit.

DATA INTEGRITY: The property that data has not been altered in an unauthorized manner. Data integrity covers data in storage, during processing, and while in transit.

DENIAL OF SERVICE: The prevention of authorized access to resources or the delaying of time-critical operations (time-critical may be milliseconds or it may be hours, depending upon the service provided).

DOMAIN: See security domain.

ENTITY: Either a subject (an active element that operates on information or the system state) or an object (a passive element that contains or receives information).

INTEGRITY: The security goal that generates the requirement for protection against either intentional or accidental attempts to violate data integrity (the property that data has not been altered in an unauthorized manner) or system integrity (the quality that a system has when it performs its intended function in an unimpaired manner, free from unauthorized manipulation).

IDENTITY: Information that is unique within a security domain and which is recognized as denoting a particular entity within that domain.

IT-RELATED RISK: The net mission/business impact considering (1) the likelihood that a particular threat source will exploit or trigger a particular information system vulnerability and (2) the resulting impact if this should occur. IT-related risks arise from legal liability or mission/business loss due to, but not limited to:

- Unauthorized (malicious, non-malicious, or accidental) disclosure, modification, or destruction of information.
- Non-malicious errors and omissions.
- IT disruptions due to natural or man-made disasters.
- Failure to exercise due care/diligence in the implementation and operation of the IT.

IT SECURITY ARCHITECTURE: A description of security principles and an overall approach for complying with the principles that drive the system design; i.e., guidelines on the placement and implementation of specific security services within various distributed computing environments.

OBJECT: A passive entity that contains or receives information. Note that access to an object potentially implies access to the information it contains.

PRINCIPLE: A rule or standard, especially of good behavior. [American Heritage Dictionary]

RISK: Within this document, synonymous with "IT-related risk."

RISK ANALYSIS: The process of identifying the risks to system security and determining the likelihood of occurrence, the resulting impact, and the additional safeguards that mitigate this impact. Part of risk management and synonymous with risk assessment.

RISK MANAGEMENT: The ongoing process of assessing the risk to mission/business as part of a risk-based approach used to determine adequate security for a system by analyzing the threats and vulnerabilities and selecting appropriate, cost-effective controls to achieve and maintain an acceptable level or risk.

SECURITY: Security is a system property. Security is much more than a set of functions and mechanisms. IT security is a system characteristic as well as a set of mechanisms that span the system both logically and physically.

SECURITY DOMAIN: A set of subjects, their information objects, and a common security policy.

SECURITY POLICY: The statement of required protection of the information objects.

SECURITY GOALS: Confidentiality, availability, integrity, accountability, and assurance.

SECURITY SERVICE: A capability that supports one, or many, of the security goals. Examples of security services are key management, access control, and authentication.

SUBJECT: An active entity, generally in the form of a person, process, or device that causes information to flow among objects or changes the system state.

SYSTEM INTEGRITY: The quality that a system has when it performs its intended function in an unimpaired manner, free from unauthorized manipulation of the system, whether intentional or accidental.

SYSTEM LIFE-CYCLE: A circular process model based on the concept that a mission need is defined and translated into an advantageous solution, which goes through a continuous loop of evolution and improvement until it is retired. There are five basic phases of the system life cycle (described in NIST Special Publication 800-14: Generally Accepted Principles and Practices for Securing Information Technology Systems):

- Initiation,
- Development/acquisition,
- Implementation,
- Operation/maintenance, and
- Disposition

THREAT: Any circumstance or event with the potential to harm an information system through unauthorized access, destruction, disclosure, modification of data, and/or denial of service. Threats arise from human actions and natural events.

THREAT SOURCE: Either (1) intent and method targeted at the intentional exploitation of a vulnerability or (2) the situation and method that may accidentally trigger a vulnerability.

THREAT ANALYSIS: The examination of threat sources against system vulnerabilities to determine the threats for a particular system in a particular operational environment.

VULNERABILITY: A weakness in system security requirements, design, implementation, or operation, that could be accidentally triggered or intentionally exploited and result in a violation of the system's security policy.

6. ADDITIONAL INFORMATION:

6.1 POLICY

Information Technology Policy 660-02: System Security

6.2 RELATED DOCUMENTS

Signed by Art Bess, Assistant Director

7. DOCUMENT HISTORY

Version	Release Date	Comments
Original	6/4/2008	